Self-organized acquisition of situated behaviors

Ralf Der
Universität Leipzig - Institut für Informatik
POB 920
D-04009 Leipzig
e-mail der@informatik.uni-leipzig.de
home page http://www.informatik.uni-leipzig.de/~der/

May 31, 2001

Abstract

The paper aims at a systematic approach to the self-organization of behavior. It is rooted in the ideas of situated artificial intelligence and introduces situated behavior as the target for the self-organization procedure. Based on a quantitative measure of behavioral situatedness a learning dynamics is introduced which enables the controller to sustain the siuatedness of the agent. The principle is demonstrated with Khepera robots in a number of different environmental conditions.

1 Introduction

The present paper is concerned with a systematic approach to self-organized behavior acquisition of embodied agents like physical robots. Our approach is rooted in the ideas of embodied cognitive science and situated Artificial Intelligence (AI). Main stream AI and cognitive science consider agents as rational beings which may exist completely separated from the real world. The world is represented by symbols and the behavior of the agent is controlled by means of logical inferences based on rules given to the agent by its designer. However this attitude insinuates a number of severe problems like the symbol grounding and the frame problem, see the discussion in [5].

Situated AI supports alternatively the conviction that intelligence is not just an abstract algorithm. Instead it emphasizes the fact that an agent which is embedded into the world in each instant may get information about the current state of the world by means of its sensors. We call an agent situated if it is able of making maximum use of its sensors in order to have the necessary information for controlling its immediate actions.

We argue that these ideas are a good starting point for a systematic approach to self-organized behavior. For this purpose we introduce the notion of (emergent) situated behavior. We call a behavior situated if the situatedness (in the above sense) of the agent while behaving is never lost. This means that control of the immediate actions and motions is organized such that after the current period of the "sense-think-act" cycle the agent is again well situated so that control is possible in the same safe way on the basis of the new (complemented by past) sensor readings. So in this diction the aim of control is control.

It is to be noted that we use the notion of situatedness in a little different sense than originally introduced by Pfeiffer. The difference is that we focus on the potential information gain the robot can achieve from the response of its sensors to its actions. Let us exemplify our idea by a "short sighted" robot (like a Khepera robot with nothing but its infrared sensors) which is to reach a door in the wall of a large room. When in empty space there is no reaction of its sensors to the motor actions. Hence all actions are of equal value and any choice of actions is purely arbitrary. However if the robot is in sensor contact to the wall, each motor action produces information in form of a definite sensor response. This information can be used for adaptation. Situated behavior would mean to follow the walls of the corridor until the door is reached.

The goal-oriented behavior of traditional AI of course is more straightforward but requires a very exact world model and dead reckoning capabilities whereas the situated behavior will work with very coarse informations about the world (numbers of doors to pass, e.g.). The wall following modus may be seen as a direct realization of our "the aim of control is control" paradigm. In fact the robot is following the wall in an ideal way if after each step the sensor situation is the same as before, i.e. the conditions of optimal control are reproduced.

The above bootstrap definition makes behavior a contingent phenomenon in contrast to the goal oriented behavior obtained from planning. Our aim is to show that this definition is immediately useful in the sense that it supports the emergence of situated behaviors in a stable and reproducible manner. Our idea is to develop a measure for the situatedness of a behavior which may serve as an objective function for the adaptation of the controller. We have developed these ideas before in the general framework of homeokinesis, cf. [3] which is the dynamical pendant of homeostasis as introduced by Cannon [2] but in the present paper we want to concentrate on the practical realization.

2 Situated behavior

Let us call $x \in \mathbf{R}^n$ the state of a system where in a concrete situation x is the vector of sensor readings of a robot. In the case of the Khepera robot $x = (x_1, ..., x_8)$ might represent the values of the eight infrared sensors. The controller of the robot may be viewed as a function mapping sensor inputs $x \in R^n$ to some outputs $y \in R^m$ governing the reaction of the robot to its specific environmental situation. The output determines the new situation (position and/or velocity) of the robot which on its hand corresponds to a new sensor input. Thus, through the sensor-motor loop, each controller produces its specific trajectory $x(t)$ in sensor space which depends both on the environment and the starting position of the robot. The evolution in time of these sensor values can be written as the update rule

$$\Delta x = \Phi\left(x; c\right)$$

where c is a set of parameters of the controller.

By way of example, let us assume for the moment that the robot is instructed to keep a fixed velocity, the controller output determining the current turn speed w of the robot in physical space. In the most simple setting we implement the controller by a single artificial neuron as [1]

$$w = \tanh\left(\sum_{i=1}^{n} c_i x_i\right) \tag{1}$$

[1] We assume that the controller does not refer to earlier sensor values as a PD controller would do. However the latter case can easily be integrated into the present concept.

where i runs over the sensors of the robot.

Figure 1: A typical environment for our robot equipped with a self-learning controller.

Each set of parameters c specifies a behavior of the robot. Our aim is to find a learning rule for the parameters c so that the robot adapts towards a situated behavior. A learning signal could be derived from a measure of the quality of situated behavior. We argue that the predictability of the sensor values over a sense-think-act cycle is a good measure in this sense. In fact if the controller manages to sustain situatedness the new sensor situation should in some sense depend in a systematic way on the old one. A pertinent example is (ideal) wall following where the new sensor values coincide with the old ones so that the prediction is trivial.

A predictor or world model may be realized by an adaptive function approximator

$$\Delta x^P = x + \psi\left(x, c; u\right)$$

where u is an additional set of parameters to be adapted such that the prediction error is minimized, i.e. the pertinent error function is

$$E = \frac{1}{2}\sum_i \left(\Delta x_i^P - \Delta x_i\right)^2$$

Δx^P being the predicted and Δx the true change in sensor values over the time step. The learning of the predictor is given by the update rule

$$\Delta u = -\eta \frac{\partial}{\partial u} E$$

In the most general case we may assume that the predictor is learned for many values of c (behaviors) so that after this learning procedure u has become a function of c. In the present paper we consider the most simple case, namely that of a trivial predictor assuming that the sensor values are constant so that

$$\Delta x^P = 0$$

In any case the error E is a function of x and c alone.

We may now use this error function and learn the behavior of the robot, i.e. update the controller parameters as

$$\Delta c = -\varepsilon \frac{\partial}{\partial c} E \tag{2}$$

which is to be executed in each time step.

We may interpret the above approach also in the terms of the complexity gap between internal (model) and external world. In all cases of practical interest the complexity of the model behavior is inferior to that of the true behavior. Thus we speak of a dynamical complexity gap between model and reality. The idea above is to use the width of the complexity gap as learning signal for the adaptation of both the model and the controller. So the aim is to adapt behavior such that the complexity gap between true and model behavior is minimized.

3 Some experimental results

In order to update c according to eq. (2) we need the gradient of E with respect to the parameters of the controller. This gradient information is gained by the following trick. We rewrite the definition of the turn velocity eq. (1) $w = \tanh(z)$ where $z = \sum c_i x_i$ and

$$\frac{\partial}{\partial c_i} E = \frac{\partial E}{\partial z} \frac{\partial z}{\partial c_i} \tag{3}$$

The last derivative can be carried out explicitly using eq. (1), so that we are left with the derivative w.r.t. to z. Let us add a small perturbation $h(t)$ to the postsynaptic potential z of the controller neuron. This may be a noise with zero mean or a periodic oscillation. Assuming that the change of h is small in the time interval between two sensor recordings, we may assume that the change in sensor values Δx and hence E is a function of the current value of h. Writing E as $E = E(x, c) = E[x, z]$ the effect of the perturbation is

$$E[x, z + \gamma h] = E + \gamma h \frac{\partial E}{\partial z} + \ldots \tag{4}$$

where $E = E[x, z]$ and γ is a strength parameter. Assume h is chosen such that in the average over a longer time interval we have $\overline{h} << \overline{h^2}$. Assuming moreover that for $h = 0$ the change of E over this period of time is sufficiently small we may write

$$\overline{Eh} \approx \gamma \overline{h^2} \frac{\partial E}{\partial z} \tag{5}$$

so that the gradient can be obtained by a time average over the product of $h(t)\, E$.

Collecting the results we have the following learning algorithm: We read the sensor values x_i at regular time intervals of length τ. The new target turn speed of the robot is defined by

$$w = \tanh\left(\sum c_i x_i + \gamma h(t)\right) \tag{6}$$

where $x_i = x_i(t)$. The update of the controller parameters is

$$\Delta c_i = -\varepsilon\, x_i\, h(t)\, E \tag{7}$$

where $E = \sum_j (\Delta x_j)^2$ and $x_i(t+\tau) - x_i(t) = \Delta x_i$. Equations (6) and (7) constitute the complete algorithm for the present robot controller.

We have implemented the above algorithm on chip of a Khepera miniature robot. In a typical experiment the parameters of the controller are initialized with $c_i = 0 \quad \forall i$. Accordingly the robot does not react to its sensor values in this initial period of time so that it will crash usually after a short time. Since its forward velocity was fixed in those experiments the robot had to be removed from the obstacle by hand. After several such trials the robot very soon learns to either avoid the wall or to follow it. The robot learns more rapidly to follow the wall, if one starts it a few times under a convenient angle to the wall. It is interesting that when following the wall for some time the controller will change the behavior mode in the sense that it finds the distance to the wall under which control is most stable. For smooth walls this means that it steers with two sensors, choosing the distance to the wall such that the sensor closest to the wall is at maximum value and the next one is at half maximum. This is a nontrivial effect and can be understood in terms of minimizing the influence of the noise as explained in [3].

The wall-following behavior remains stable as long as there are no dramatic changes in the environment. This could be corroborated in a long time experiment where the robot was running nearly three days completely autonomously in a simple maze. It should be mentioned that during all experiments there was no tuning of the learning parameters, in particular we did not use any “cooling” of the learning rate ϵ.

A further point of general interest is that the robot changes its behavior depending on the environmental conditions. Typically one can learn it to

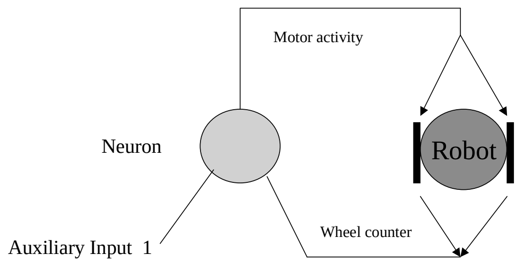

Figure 2: A neuron controlling the forward velocity of a robot.

balance a ball and than put the learned robot under some angle close to the wall. Of course the robot will show the same reaction as with a ball, i.e. it turns until the wall is in front of it and then bangs against the wall. However after helping it several times out of this crash situation it will learn to follow the wall afterwards.

It is also interesting that the controller may cope with the (simulated) damage of a sensor as demonstrated with the ball balancing behavior which the robot learns even if one of the sensors involved is damaged.

4 Emergent situated behavior in a basic sensor-motor loop

In the above scenario we have controlled only the turn velocity of the robot. Driven by a fixed forward velocity the robot controller learns to follow walls and/or to avoid obstacles since in this way it manages to keep its sensor values as constant as possible. Is there a similar approach to the control of the forward velocity of the robot? Let us consider a neuron as an adaptive agent embedded into the sensor-motor loop. The neuron is to control the velocity of the robot, i.e. its output is to be the target velocity v and its input the true velocity of the wheels as read back from the wheel counters. Hence the dynamics of this sensor-motor loop is given by

$$v_{n+1} = \tanh\left(cv_n + H\right) + \xi_n \tag{8}$$

where H may be interpreted as a threshold and ξ_n is the noise due to the difference of the target and the true velocity of the wheels caused by slip and friction effects. Without noise the dynamics formulated by eq. (8) for any set of parameters always converges to a fixed point where v does not change any more (the robot moves with constant velocity), the fixed point equation reading

$$v = \tanh\left(cv + H\right) \qquad (9)$$

On the other hand for each value of v where $|v| < 1$ there is a one parameter set of values c, H so that this v is a fixed point of the above dynamics. In particular for $H = 0$ there is a stable fixed point at $v = 0$ for all values $0 < c < 1$ whereas for $c > 1$ there are two stable fixed points at $v = \pm q$ where q goes from 0 to 1 for increasing values of $c > 1$.

Being at the fixed point corresponds to a situated behavior with optimal control of the situation. However the noise interferes with this convenient situation and we formulate a learning rule for the reinstallation of situatedness after each perturbation by the noise. According to eq. (9) the appropriate error function is

$$F = \frac{1}{2}\left(\tanh\left(cv_n + H\right) - v_n\right)^2$$

the error being zero if v_n is a fixed point. Correspondingly the learning rule is

$$\begin{aligned} \Delta c &= -\varepsilon\left(\tanh\left(cv_n + H\right) - v_n\right)\, Q\, v_n \qquad (10) \\ \Delta H &= -\varepsilon\left(\tanh\left(cv_n + H\right) - v_n\right)\, Q \end{aligned}$$

where v_n is the current value of the velocity as read by the wheel counters and $Q = \tanh'\left(cv_n + H\right) = 1 - \tanh^2\left(cv_n + H\right)$ is the derivative of the tanh function.

The combined dynamics given by eqs. (8) and (10) displays very interesting properties. Driven by the noise the system visits its fixed points in a more or less erratic way. When starting with $c = 0$ and $H = 0$ the values of v visited scatter around $v = 0$ while c is increasing steadily. After sufficient time the value of c stabilizes at about $c = 1.2$ while H fluctuates around zero. With these parameters the v dynamics is characterized by staying predominantly in the symmetry broken states at $v = \pm q$ with $q \approx 0.7$ with v switching between these values in an irregular mode.

If the mean value of the noise is zero, a robot controlled by this neuron will move forward for some time and then reverse its velocity and so on, the distances covered by this erratic motion depending on the strength of the noise and the learning rate in a systematic way. For the Khepera robot the typical distance coveed in the unidirectional motion is about 1 to 5 m. The most interesting property of this controller paradigm however is observed if the robot collides with some obstacle. In this case the noise (difference between true and target velocity) is biased which leads to a very rapid relearning of the parameters c and H such that the velocity of the robot is reversed almost immediately.

In this way our learning dynamics may be said to generate an explorative behavior of the robot with a sensitive reaction to perturbations from the environment. These ideas may be seen in a more general context. The time discrete stochastic dynamical system given by eq. (8) is symmetric with respect to the transformation $v \to -v$ and $H \to -H$. For $c > 1$ ($H = 0$ for simplicity) the symmetric solution $v = 0$ is destabilized in favor of the symmetry broken solutions $v = \pm q$ with $q > 0$. The interesting point is that the learning dynamics drives the system towards this symmetry breaking scenario c however staying in a region where the system still may jump between the two branches of the bifurcation diagram. This is what keeps the system sensitive to any bias in the noise.

The observed properties may be called a situated behavior in empty space. In fact, situatedness means being rooted by the sensors to the structures and dynamics of the world. In empty space the only information is from internal sensors (not present in the Khepera robot) and the wheel sensors feeling the motion of the robot. Now assume the unlearned controller generates a stochastic motion. The information the robot will gather is that the world is invariant w.r.t. translations, rotations, and time reversal. In this sense we might say the space is not empty but full of symmetries. A situated behavior in empty space is defined by (dynamical) relations to these symmetries. The two possible relations are either (i) to obey the symmetries which in the present case means to stay in the state $v = 0$ or (ii) to break the symmetries which is what the robot learns to do under the above learning dynamics. The robot arrives at executing dynamical patterns (search patterns) in space.

5 Conclusions

Our experiments with a Khepera robot clearly demonstrated that the aim of sustaining situatedness is a fertile starting point for the self organization of behavior. Depending on the environmental conditions [2] we have observed a number of different behaviors with always the same adaptive controller and learning parameters. In particular we observed behaviors like collision avoidance, ball balancing, navigation through narrow corridors, different modi of wall following, and navigational capabilities in chaotic mazes. In many situations we observed that the controller converged towards the best working conditions, i.e. the behavior where control is most safely realized. The latter effect results from a nontrivial combination of nonlinearity and noise effects.

[2] The environmental conditions include the human interferences like choosing the starting position of the robot and the way of restarting it after helping it out of a crash situation.

Form the practical point of view the learning dynamics for the neuron controlling the forward velocity, cf. eq. (10), proves very helpful. Due to the sensitive reaction of the neuron to "slip and friction noise" our robot survives in nearly arbitrary environments without getting stuck in corners or at other obstacles. In this way the turn velocity controller is provided with the informations necessary for the acquisition of a situated behavior based on the infrared sensors. Moreover, learning is found to be extremely fast and permanently alert while under stationary environmental conditions it is convergent and reproducible.

Our future work will be devoted to testing the current controller under new environmental conditions while including the camera and the ambient light sensors. We will also study emerging collective behaviors with several robots in a common environment each one being controlled by our adaptive controller. The main challenge however is the integration of the emerging behaviors into an intentional behavior based controller architecture [1] and into the scenario of evolutionary robotics[4].

References

[1] R. C. Arkin. *Behavior Based Robotics.* MIT Press, 1998.

[2] W. B. Cannon. *The wisdom of the body.* Norton, New York, 1939.

[3] R. Der, U. Steinmetz, and F. Pasemann. Homeokinesis - a new principle to back up evolution with learning. In M. Mohammadian, editor, *Computational Intelligence for Modelling, Control, and Automation*, volume 55 of *Concurrent Systems Engineering Series*, pages 43 – 47. IOS Press, 1999. URL=http://www.informatik.uni-leipzig.de/ der/Veroeff/wienfin3.ps.

[4] S. Nolfi and D. Floreano. *Evolutionary Robotics.* MIT, 2000.

[5] R. Pfeiffer and C. Scheier. *Understanding Intelligence.* MIT Press, 1999.